

Appl. Serial No. 09/802,875  
Amendment dated October 5, 2004  
Reply to Office Action of May 5, 2004

**Amendments to the Specification**

Page 2, cancel lines 4-9, and substitute the following paragraph therefor:

Accordingly, a primary object of the present invention is to provide an improved method for designing a rail-mounted component assembly for an electrical power distribution system, wherein a software routine is utilized to simulate the in-line assembly on a computer screen, with the individual junction blocks or terminals being depicted directly as the image of a data structure consisting of individual token or replica elements mounted on a support rail.

Page 2, line 21, to Page 3, line 4, cancel the paragraph, and substitute the following paragraph therefor:

According to another object of the invention, the terminal blocks and/or other elements and/or other equipment are illustrated on the screen in a photorealistic manner. In the process, the individual clamp elements are stored in a databank and, during simulation of the locking of the terminal block on the mounting rail. Preferably, a memory area is reserved in a memory for a product to be illustrated, whereupon, necessary information will be read out of a databank, will be interpreted, and will be displayed as the smallest possible graphic element on the screen, with each individual element representing one of the tokens. In the process, each token replica is preferably allocated changeable properties.

Page 7, cancel lines 10-25, and substitute the following paragraphs therefor:

The operation of the invention will be explained in greater detail. According to the invention, the products - such as, for instance, a terminal block - are illustrated graphically in several steps. First of all, one reserves the necessary memory space for the project to be illustrated. This is followed by the detailed construction of the partial

Appl. Serial No. 09/802,875  
Amendment dated October 5, 2004  
Reply to Office Action of May 5, 2004

elements in this framework, from top to bottom. For this purpose, one reads the necessary information items out of a databank. Then, these information items are interpreted and are displayed on the screen as the smallest possible graphic element, as shown in Fig. 1. These individual graphic elements are referred to below as tokens or replicas. The number of necessary tokens replicas to illustrate a product depends on the complexity of the product. The type, size, and appearance of a token replica; that is, the properties of the element can be modified as desired by various methods. The sum of all tokens replicas forms the complete illustration of the product on the screen. Every token replica can take on any shape; it is therefore possible to illustrate any products on the screen and to provide them with active functional qualities. All tokens replicas have properties that can be changed by means of corresponding methods. This might be explained in greater detail with the aid of an illustrative example.

Page 8, cancel lines 14-21, and substitute the following paragraphs therefor:

The above-illustrated procedure is analogous to the procedure involved in the graphic illustration of the products on the screen in the context of the present invention. The tokens replicas basically correspond to the cardboard pieces. The type, size, and appearance of the token replica; that is, the properties of the element can be modified as desired by different methods. The sum of the tokens replicas forms the illustration of the product (for example, terminal block) on the screen. An example here can be found in Fig. 4. This is a simplified illustration of a terminal block with its tokens replicas and their properties.

Page 9, cancel lines 3-21, and substitute the following paragraph therefor:

On the other hand, the main advantage inherent in the system of the present invention is that it can be put together with the graphic data which therefore can be processed in a particularly fast manner. Furthermore, it is possible to adapt the properties

Appl. Serial No. 09/802,875  
Amendment dated October 5, 2004  
Reply to Office Action of May 5, 2004

of a product to individual needs. There is no need for any further programming to illustrate new surfaces. Merely by supplementing the databank with new articles and the pertinent documents, one can now visually display new designs (see Figs. 1 and 2). Fig. 2 shows an article selection window. By specifically putting in search criteria, one can, by means of this search window, find the desired article and it can be included in the table of the article library for further processing of the project. This procedure entails the advantage that the illustration of various products can take place quickly and realistically on the screen. It is furthermore also possible to modify a product, whose illustration has been finished, on the screen, via the user surface. Each product, even in detail, has an intelligence of its own that is represented by the token replica. The memory requirement, compared to the picture formats of the kind mentioned above, is very small; this has yet another advantageous effect as regards the processing speed. As an example for an intelligence bearing token replica, one might refer to Fig. 4. In the upper example, the token replica is illustrated, whereas, in the lower example, the token replica is provided with a marker with any desired text information to illustrate the terminal block.

Page 10, cancel lines 1-7, and substitute the following paragraph therefor:

When tokens replicas are used, on the other hand, the marker token replica is merely informed that the color and text content must be changed. It is thus easily and without any problems possible to put in and immediately display any text information. The reverse is also possible. The token replica can, in the user program, at any time, communicate its current properties. In cross-product functions, for example, when putting in cross connections, the X, Y, and Z positions of a token replica, can be used, for instance, for a functional check.